CIS 705 — Programming Languages — Spring 2009

Assignment 2

Due by 2:30 p.m. on Thursday, February 26

The context for this assignment is Chapter 2 of *DAWOC* and
the core and standard OCaml libraries.

Definitions

- The *sum* of a list of integers xs is the result of adding together all of the elements of xs (yielding 0, if xs is empty). Here, as elsewhere in this assignment, you can ignore the possibility of overflow.

- A list xs is a *sublist* of a list ys iff there are lists us and vs such that $us \,@\, xs \,@\, vs = ys$.

- A list xs is a *proper sublist* of a list ys iff xs is a sublist of ys, and $xs \neq ys$.

- A list of integers xs is *balanced* iff

    - the sum of xs is 0,

    - xs is nonempty and

    - no proper, nonempty sublist of xs has a sum of 0.

Finding Balanced Sublists

The goal of this assignment is to write an OCaml program for finding the balanced sublists of a list of integers. You should define a function

```
val balanced : int list -> int list list
```

such that `balanced` xs is the list of all balanced sublists of xs, listened in strictly ascending order according to OCaml's standard ordering (as implemented by `compare` and `<`).

For example, `balanced[1;1;-1;1]` should evaluate to `[[-1;1];[1;-1]]`.

Assessment

Your program will be assessed according to three criteria:

- **Correctness**. Test your program carefully, but also try to prove to yourself that it is correct. (You don't have to submit evidence of testing, or to write your proof down.)

- **Style**. Program in a strictly functional style, without using the imperative features of Chapter 3 of *DAWOC*. Format your program in a way that makes its structure clear. Choose meaningful names for functions, and choose other identifiers with care. Document your functions by abstractly explaining their input/output behavior. (For a given function, say that if we know that its input has some property, that its output will have some other property).

- **Efficiency**. Try to make your program efficient, but don't sacrifice correctness or style in doing so.

Submission

Begin your program with a comment containing your name. Submit your program by emailing it to me. I will acknowledge receiving it. Make sure that you retain an electronic copy of your program.